

体系结构描述文档

灯具照明行业进销存管理系统



2014-11-3

小百合工作室

# 更新历史

|  |  |  |  |
| --- | --- | --- | --- |
| 修改人员 | 日期 | 变更原因 | 版本号 |
| 陈云龙 | 2014-11-03 | 初步文档框架 | V0.0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[更新历史 1](#_Toc402881706)

[1. 引言 1](#_Toc402881707)

[1.1. 编制目的 1](#_Toc402881708)

[2. 产品概述 1](#_Toc402881709)

[3. 体系结构设计概述 1](#_Toc402881710)

[4. 结构视角 1](#_Toc402881711)

[4.1. 业务逻辑层的分解 1](#_Toc402881712)

[4.2. 数据层的分解 2](#_Toc402881713)

[5. 依赖视角 2](#_Toc402881714)

# 引言

## 编制目的

本报告详细完成灯具照明行业进销存系统的详细设计，达到指导后续软件构造的目的，同时实现和测试人员及用户的沟通。

本报告面向开发人员、测试人员及最终用户而编写，是了解系统的导航。

# 产品概述

参考灯具照明行业进销存系统用例文档和灯具照明行业进销存系统软件规格说明中对产品的概括描述。

# 体系结构设计概述

请参考进销存系统需求文档对体系结构设计的概述

# 结构视角

## 业务逻辑层的分解

业务逻辑层的开发包图详见体系结构文档图3。

* + 1. Commoditybl
    2. CommoditySortbl
    3. Inventorybl
    4. Clientbl
    5. Purchasebl
    6. Salebl
    7. Recordbl
    8. Accountbl
    9. AccountBillbl
    10. CashBillbl
    11. Accountainit
    12. Approvalbl
    13. Promotionbl
    14. Userbl

## 数据层的分解

数据层的开发包图请参见体系结构描述文档图4。

### DataFactoryService

1. 模块概述

DataFactoryService模块是一个工厂类，用来创建其他的DataService实例，并将其统一管理起来。

1. 整体结构

DataFactoryService模块设计如下图所示



1. 模块内部的接口规范

DataFactory内部的的接口规范如下表

**DataFactory的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| DataFactory.getAccountaInitData | 语法 | **public DataService getAccountaInitData() throws RemoteException** |
| 前置条件 | 无 |
| 后置条件 | 返回**AccountaInitData**对象 |
| DataFactory. getAccountData | 语法 | **public DataService getAccountData() throws RemoteException;** |
| 前置条件 | 无 |
| 后置条件 | 返回**AccountData**对象 |
| DataFactory. getApprovalData | 语法 | **public DataService getApprovalData() throws RemoteException;** |
| 前置条件 | 无 |
| 后置条件 | 返回**ApprovalData**对象 |
| DataFactory. getClientData | 语法 | **public DataService getClientData() throws RemoteException;** |
| 前置条件 | 无 |
| 后置条件 | 返回**ClientData**对象 |
| DataFactory. getCommodityData | 语法 | **public DataService getCommodityData() throws RemoteException;** |
| 前置条件 | 无 |
| 后置条件 | 返回**CommodityData**对象 |
| DataFactory.  getCommoditySortData | 语法 | **public DataService getCommoditySortData () throws RemoteException;** |
| 前置条件 | 无 |
| 后置条件 | 返**回CommoditySortData**对象 |
| DataFactory. getAccountBillData | 语法 | **public DataService getAccountBillData () throws RemoteException** |
| 前置条件 | 无 |
| 后置条件 | 返回**AccountBillData**对象 |
| DataFactory. getInventoryData | 语法 | **public DataService getInventoryData () throws RemoteException** |
| 前置条件 | 无 |
| 后置条件 | 返回**InventoryData**对象 |
| DataFactory. getPromotionData | 语法 | **public DataService getPromotionData () throws RemoteException** |
| 前置条件 | 无 |
| 后置条件 | 返回**PromotionData**对象 |
| DataFactory. getPurchaseData | 语法 | **public DataService getPurchaseData () throws RemoteException** |
| 前置条件 | 无 |
| 后置条件 | 返回**PurchaseData**对象 |
| DataFactory. getRecordData | 语法 | **public DataService getRecordData () throws RemoteException** |
| 前置条件 | 无 |
| 后置条件 | 返回**RecordData**对象 |
| DataFactory. getSaleData | 语法 | **public DataService getSaleData () throws RemoteException** |
| 前置条件 | 无 |
| 后置条件 | 返回**SaleData**对象 |
| DataFactory. getUserData | 语法 | **public DataService getUserData () throws RemoteException** |
| 前置条件 | 无 |
| 后置条件 | 返回**UserData**对象 |
| DataFactory. getCashBillData | 语法 | **public DataService getCashBillData () throws RemoteException** |
| 前置条件 | 无 |
| 后置条件 | 返回**CashBillData**对象 |

### InventoryDataService

1. 模块概述

InventoryDataService模块提供库存单据的提交服务（赠送单、报溢单、报损单、报警单）

1. 模块内类接口规范

InventoryDataService接口规范如下表

**InventoryDataService接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| InventoryDataService.insert | 语法 | public ResultMessage insert(InventoryBillPO po) |
| 前置条件 | 无 |
| 后置条件 | 返回提交是否成功 |
| InventoryDataService.find | 语法 | public InventoryBillPO find(String ID) |
| 前置条件 | 无 |
| 后置条件 | 查找返回相应的结果 |
| InventoryDataService.update | 语法 | public ResultMessage update (InventoryBillPO po) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 更新po |
| InventoryDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### PurchaseDataService

1. 模块概述

PurchaseDataService提供进货（进货退货）数据集体载入、保存、增加、删除、查找

1. 模块内部类的接口规范

PurchaseDataService类的接口规范如下表所示

**PurchaseDataService类接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| PurchaseDataService.insert | 语法 | public ResultMessage insert(PurchasePO po) |
| 前置条件 | 同样的po在数据中不存在 |
| 后置条件 | 增加一条po数据 |
| PurchaseDataService.find | 语法 | public PurchasePO find(String id) |
| 前置条件 | 无 |
| 后置条件 | 查找返回相应的结果 |
| PurchaseDataService.delete | 语法 | public ResultMessage delete (String ID) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 删除po |
| PurchaseDataService.update | 语法 | public ResultMessage update (PurchasePO po) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 更新po |
| PurchaseDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### RecordDataService

1. 模块概述

RecordDataService提供销售明细表、经营历程表、经营情况表的查看

1. 模块内类接口规范

RecordDataService类的接口规范如下表所示

**RecordDataService接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| RecordDataService.getSaleDetail | 语法 | public ArrayList<PersistentObject> getSaleDetail  (String item) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 返回符合上述条件的商品销售记录，包含如下信息：时间（精确到天），商品名，型号，数量，单价，总额 |
| RecordDataService.getBusinessProcess | 语法 | public ArrayList<PersistentObject> getBusinessProcess  (String item) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 返回1. 销售类单据（销售出货单，销售退货单） 2. 进货类单据（进货单，进货退货单） 3. 财务类单据（付款单，收款单，现金费用单）4. 库存类单据（报溢单，报损单，赠送单）。 |
| RecordDataService.getBusinessCondition | 语法 | public ArrayList<PersistentObject> getBusinessCondition  (String item) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 返回信息： 1. 收入类：销售收入、商品类收入（商品报溢收入 成本调价收入 进货退货差价 代金券与实际收款差额收入）。收入类显示折让后总收入，并显示折让了多少。2. 支出类：销售成本、商品类支出（商品报损 商品赠出）。支出类显示总支出。3. 利润：折让后总收入-总支出 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### AccountBillDataService

1. 模块概述

AccountBillDataService提供收款单、付款单、现金单的集体载入、保存、查看修改操作

1. 模块内类的接口规范

AccountBillDataService类的接口规范如下表所示

**AccountBillDataService类的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| FundDataService.insert | 语法 | public ResultMessage insert  (AccountBillPO po) |
| 前置条件 | 同样的po在数据中不存在 |
| 后置条件 | 增加一条po数据 |
| FundDataService.find | 语法 | public AccountBillPO find(String ID) |
| 前置条件 | 无 |
| 后置条件 | 查找返回相应的结果 |
| FundDataService.delete | 语法 | public ResultMessage delete  (String ID) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 删除po |
| FundDataService.update | 语法 | public ResultMessage update  (AccountBillPO po) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 更新po |
| FundDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### ApprovalDataService

1. 模块概述

ApproalDataService提供需要审批的单据的集体载入、保存、查看修改操作

1. 模块内部类的接口

**ApproalDataService接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| ApprovalDataService.update | 语法 | public ResultMessage update(PersistentObject po) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 更新po |
| ApprovalDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### UserDataService

1. 模块概述

提供用户集体载入、保存、增删改查服务

1. 模块内部类的接口

**UserDataService接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| UserDataService.insert | 语法 | public ResultMessage insert(UserPO po) |
| 前置条件 | 同样的po在数据中不存在 |
| 后置条件 | 增加一条po数据 |
| UserDataService.find | 语法 | public UserPO find(String username) |
| 前置条件 | 无 |
| 后置条件 | 查找返回相应的结果 |
| UserDataService.delete | 语法 | public ResultMessage delete (String ID) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 删除po |
| UserDataService.update | 语法 | public ResultMessage update (UserPO po) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 更新po |
| UserDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### CommodityDataService

1. 模块概述

提供商品集体载入、保存、增删改查服务

1. 模块内部类的接口

**CommodityDataService接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| CommodityDataService.insert | 语法 | public ResultMessage insert(CommodityPO po) |
| 前置条件 | 同样的po在数据中不存在 |
| 后置条件 | 增加一条po数据 |
| CommodityDataService.find | 语法 | public ResultMessage find(String name) |
| 前置条件 | 无 |
| 后置条件 | 查找返回相应的结果 |
| CommodityDataService.delete | 语法 | public ResultMessage delete (String ID) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 删除po |
| CommodityDataService.update | 语法 | public ResultMessage update (CommodityPO po) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 更新po |
| CommodityDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### CommoditySortDataService

1. 模块概述

提供商品分类数据集体载入、保存、增删改查服务

1. 模块内部类的接口

**CommoditySortDataService接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| CommoditySortDataService.insert | 语法 | public ResultMessage insert  (CommoditySortPO po) |
| 前置条件 | 同样的po在数据中不存在 |
| 后置条件 | 增加一条po数据 |
| CommoditySortDataService.find | 语法 | public CommoditySortPO find(String name) |
| 前置条件 | 无 |
| 后置条件 | 查找返回相应的结果 |
| CommoditySortDataService.delete | 语法 | public ResultMessage delete  (String id) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 删除po |
| CommoditySortDataService.update | 语法 | public ResultMessage update  (CommoditySortPO po) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 更新po |
| CommoditySortDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| CommoditySortDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### SaleDataService

1. 模块概述

提供销售（销售退货）数据集体载入、保存、增加、删除、查找

1. 模块内部类的接口

**SaleDataService的接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| SaleDataService.insert | 语法 | public ResultMessage insert(SalesPO po) |
| 前置条件 | 同样的po在数据中不存在 |
| 后置条件 | 增加一条po数据 |
| SaleDataService.find | 语法 | public SalesPO find(String id) |
| 前置条件 | 无 |
| 后置条件 | 查找返回相应的结果 |
| SaleDataService.delete | 语法 | public ResultMessage delete (String id) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 删除po |
| SaleDataService.update | 语法 | public ResultMessage update (SalesPO po) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 更新po |
| SaleDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### AccountDataService

1. 模块概述

提供账户数据集体载入、保存、增删改查服务

1. 模块内部类的接口

**AccountDataService接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| AccountDataService.insert | 语法 | public ResultMessage insert(AccountPO po) |
| 前置条件 | 同样的po在数据中不存在 |
| 后置条件 | 增加一条po数据 |
| AccountDataService.find | 语法 | public AccountPO find(String name) |
| 前置条件 | 无 |
| 后置条件 | 查找返回相应的结果 |
| AccountDataService.delete | 语法 | public ResultMessage delete (String name) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 删除po |
| AccountDataService.update | 语法 | public ResultMessage update (AccountPO po) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 更新po |
| AccountDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### ClientDataService

1. 模块概述

提供客户数据集体载入、保存、增删改查服务

1. 模块内部类的接口

**ClientDataService类接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| ClientDataService.insert | 语法 | public ResultMessage insert(ClientPO po) |
| 前置条件 | 同样的po在数据中不存在 |
| 后置条件 | 增加一条po数据 |
| ClientDataService.find | 语法 | public ClientPO find(String name) |
| 前置条件 | 无 |
| 后置条件 | 查找返回相应的结果 |
| ClientDataService.delete | 语法 | public ResultMessage delete (String id) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 删除po |
| ClientDataService.update | 语法 | public ResultMessage update (ClientPO po) |
| 前置条件 | 同样的po在数据中存在 |
| 后置条件 | 更新po |
| ClientDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### AccountaInitDataService

1. 模块概述

提供期初建账数据集体载入、保存、增加、查看服务

1. 模块内部类的接口

**AccountaInitDataService接口规范**

|  |  |  |
| --- | --- | --- |
| 提供的服务（供接口） | | |
| AccountaInitDataService.insert | 语法 | public ResultMessage insert(AccountaInitPO po) |
| 前置条件 | 同样的po在数据中不存在 |
| 后置条件 | 增加一条po数据 |
| AccountaInitDataService.init | 语法 | public void init() |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |
| AccountaInitDataService.getAllMsg | 语法 | public ArrayList<AccountaInitPO> getAllMsg (); |
| 前置条件 | 无 |
| 后置条件 | 返回全部AccountaInitPO持久数据 |
| InventoryDataService. getID | 语法 | public void getID () |
| 前置条件 | 无 |
| 后置条件 | 初始化持久数据 |

### PromotionDataService

1. 模块概述
2. 模块内部类的接口

# 依赖视角

下图是客户端和服务器端各自的包之间的依赖关系



图 1 客户端包图

图 2 服务器端包图